ASSIGNMENT 2

(Javascript Arrays,Logical Statements,Control Flow and OOPs Concept)

Q 1. Manipulating an array
Take the following array:
const theList = ['Laurence', 'Svekis', true, 35, null, undefined,
{test: 'one', score: 55}, ['one', 'two']];
Manipulate your array using various methods, such as pop(), push(), shift(), and unshift(), and transform it into the following:
["FIRST", "Svekis", "MIDDLE", "hello World", "LAST"]
Take the following steps, or adopt your own approach:
• Remove the first item and the last item.
• Add FIRST to the start of the array.
• Assign hello World to the fourth item value.
• Assign MIDDLE to the third index value.
• Add LAST to the last position in the array.
• Output it to the console.

Q 2. Company product catalog
In this project, you will implement a data structure for a product catalog and create queries to retrieve data.

- Create an array to hold an inventory of store items.
- Create three items, each having the properties of name, model, cost, and quantity.
- Add all three objects to the main array using an array method, and then log the inventory array to the console.
- Access the quantity element of your third item, and log it to the console. *Experiment by adding and accessing more elements within your data Structure.*

Q 3. Friend checker game
Ask the user to enter a name, using the switch statement to return a confirmation that the user is a friend if the name selected is known in the case statements. You can add a default response that you don't know the person if it's an unknown name. Output the result into the console.

Q 4. Rock Paper Scissors game

This is a game between a player and the computer, where both will make a random selection of either Rock, Paper, or Scissors (alternatively, you could create a version using real player input!). Rock will beat out Scissors, Paper will beat out Rock, and Scissors will beat out Paper. You can use JavaScript to create your own version of this game, applying the logic with an if statement. Since this project is a little more difficult, here are some suggested steps:

1. Create an array that contains the variables Rock, Paper, and Scissors.
2. Set up a variable that generates a random number 0-2 for the player and then do the same for the computer's selection. The number represents the index values in the array of the 3 items.
3. Create a variable to hold a response message to the user. This can show the random results for the player and then also the result for the computer of the matching item from the array.
4. Create a condition to handle the player and computer selections. If both are the same, this results in a tie.
5. Use conditions to apply the game logic and return the correct results.There are several ways to do this with the condition statements, but you could check which player's index value is bigger and assign the victory accordingly, with the exception of Rock beating Scissors.
6. Add a new output message that shows the player selection versus the computer selection and the result of the game.

Q 5. Math multiplication table

In this project, you will create a math multiplication table using loops. You can do this using your own creativity or by following some of the following suggested steps:

1. Set up a blank array to contain the final multiplication table.
2. Set a value variable to specify how many values you want to multiply with one another and show the results for.
3. Create an outer for loop to iterate through each row and a temp array to store the row values. Each row will be an array of cells that will be nested into the final table.
4. Add an inner for loop for the column values, which will push the multiplied row and column values to the temp array.
5. Add the temporary row data that contains the calculated solutions to the main array of the final table. The final result will add a row of values for the calculations.

Q 6. Create a recursive function

Create a recursive function that counts up to 10. Invoke the function with different start numbers as the arguments that are passed into the function. The function should run until the value is greater than 10.

Q 7. Set timeout order

Use the arrow format to create functions that output the values one and two to the console. Create a third function that outputs the value three to the console, and then invokes the first two functions. Create a fourth function that outputs the word four to the console and also use setTimeout() to invoke the first function immediately and then the third function.

What does your output look like in the console? Try to get the console to output:
Four Three One Two One

Q 8. Employee tracking app

Create a class to track the employees of a company:

- Use first names, last names, and the number of years worked as values in the constructor.
- Create two or more people with values for their first names, last names, and the number of years they've worked at the company. Add the people into an array.
- Set up a prototype to return the details of the person's first and last names and how long they've worked at the company.
- Iterate the contents of the array to output the results into the console, adding some text to make the output a full sentence.

Q 9. Menu items price calculator

Create a class which will allow you to work out the combined price of a number of items, and interact with it to work out the total cost of different orders.

- Create a class that contains the prices of two menu items as private field declarations.
- Use the constructor in the class to get the argument values (how many of each item are being bought).
- Create a method to calculate and return the total cost depending on how many of each item the user selects.
- Use a getter property to grab the value output by the calculation method.
- Create two or three objects with different combinations of menu selections, and output the total cost in the console.